

Weekly Report (2017.9.4-2017.9.10)

TASK	DEADLINE	CURRENT PROGRESS
Visual analysis system of LAN	8.31	<p>It took a lot of time to upload different kinds of files. Consider design, access and connectivity of the database.</p> <p>Now the problem is that the project has been overdue, but there are many search and interaction of the graphs need to be done.</p> <p>And current proposed solution is that make correct assessment and plan, and be more efficient.</p>

Done

1) Add and modify the page styles.

Implement the search box, and add toggle animation to it.

Implement the bubble box for search box's frame. The way to realize it is as follow:

<http://www.cnblogs.com/daxiong/articles/3158630.html>

The realization of the information table. At the very beginning, I plan to code it myself, just implementing what we need. Dr. Mei reminds me that the realization of the table interaction is very complicated. So I decide to use the existing plugin---HandsonChart which has almost all what we want.

2) Something about database.

The database table field which is set to be incremented must be set as the primary key. The SQL statement can be written as follow:

```
CREATE TABLE IF NOT EXISTS computer (  
id INT(10) NOT NULL AUTO_INCREMENT,  
name LONGTEXT,  
PRIMARY KEY(id)  
);
```

The SQL statement to join two database table in the sake of searching can be as follow:

```
select * from dns,relationship where relationship.IP = dns.IP AND  
relationship.domainName='cosmos'
```

The reference is: <http://blog.csdn.net/liuchunming033/article/details/47420981>

3) Some work about uploading different kinds of files and write new files.

When we upload files, we first read the origin files, and then the stream exit in two ways---writing into new files and inserting into database.

In terms of writing into new files, I use fast-csv library. Create different kinds of new file by the means of adding new pipe. If I need to append data to a existing file instead of creating, just change a option attribute of pipe like this demo:

```
var writeStream = fs.createWriteStream(csvPath, {  
  encoding: "utf8",  
  flags: "a"  
});
```

Change the default “w” flags to “a” flags.

Both writing files and inserting into database are asynchronous. And when it comes to asynchronous, it usually accompanied by a lot of problems.

Firstly, the original file data don't have the field ID, so I add this field in the meantime of inserting into database. However, I also need to write this field into new file. So I need to obtain ID after database inserting. This is a asynchronous process, and what's worse, I can't obtain the ID. I tried nested way and Promise gramma, both are failed. At last, I generate the ID using guid before inserting into database instead of the auto-incremental ID.

Because of the original file is quite large, the error that Javascript heap out of memory appears sometimes. When this kind of error occurs, it means that I need to modify the implement logic. Quite annoying thing taking much time.

Some complicated searching task, such as two layers of nested queries, also a asynchronous problem. The debug takes a lot of time. Finally I use Promise.all to fix it. Just as follow:

```
function promiseQuery(sql, params) {
  return new Promise(function(resolve, reject) {
    db.query(sql, params, function(result) { resolve(result); })
  })
}

promiseQuery(
  'select diaryTime,id as userId,userName,IP from relationship,' + csvTable + '
where relationship.userName = ' + csvTable + '.name AND relationship.domainName=?',
  [domain])
  .then(function(data) {
    Promise.all(data.map(function(d) {
      return promiseQuery('select computerName,id as computerId from dnsdata,
oozzoocsv where dnsdata.computerName=oozzoocsv.name and dnsdata.IP=?', d.IP);
    })).then(function(results) {
      res.body = results;
    })
  })
});
```

To Do

- 1) Finish the development of system as fast as I can.
- 2) Read the paper related to Vega Lite.

HAPPY TEACHERS' DAY!